

# RELATÓRIO DE ATIVIDADES DA IC 16-12-2024

João Lucas Santos Penha de Oliveira

Em primeiro momento, foi realizado a comunicação via wifi entre as duas ESP32, neste tipo de comunicação, é preciso conectar cada dispositivo a uma rede.

Para isto existem duas possibilidades mais utilizadas, a primeira é conectar as duas ESP32 a uma rede hospedada por um roteador, e elas se conectar a partir do IP obtida ao se conectar na rede, a segunda possibilidade é fazer com que uma ESP32 hospede a rede wifi, e a outra então se conecta. A grande vantagem da segunda modalidades, é da não necessidade de usar um outro dispositivo para conectar as duas ESP32, e por conta disso, esta foi optada para ser usado no projeto.

Em resumo, uma ESP32 cria uma nova rede wifi, com um nome a escolha, e com uma senha a escolha, neste caso foi utilizado o nome “ESP32-Master” e a senha “123456789”. Com a rede criada, a segunda ESP32 se conecta nesta rede a partir dessas credenciais.

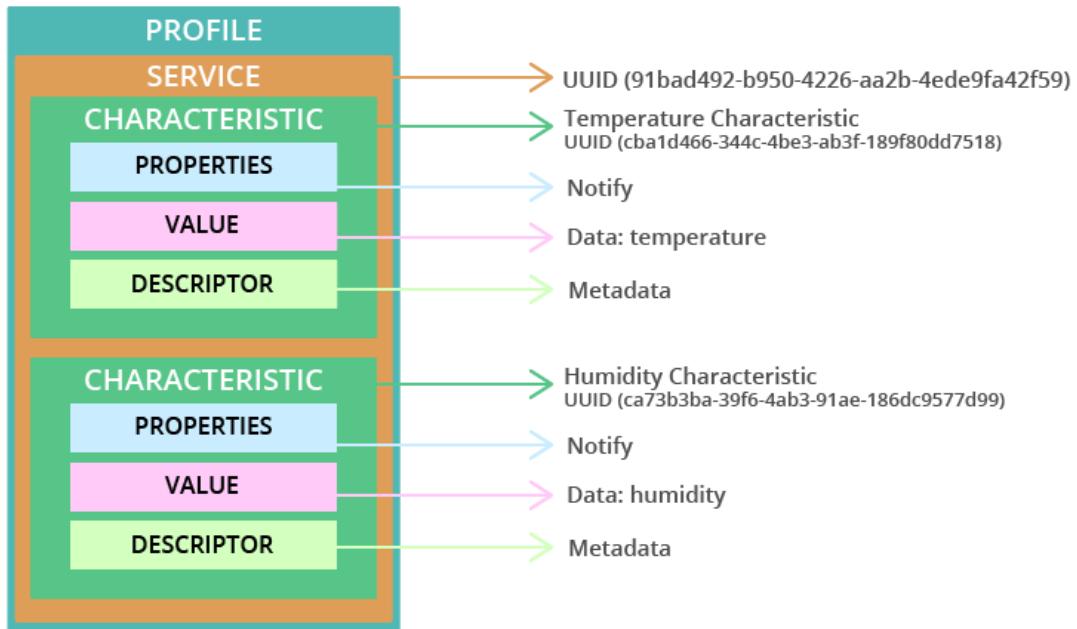
Com ambas conectadas, o próximo passo é simplesmente mandar uma mensagem de uma para outra. Para isto, o padrão escolhido é que sempre a ESP32 Slave envia a mensagem primeiro, e então a ESP32 Master retorna com uma mensagem com outra mensagem. Este tipo de padrão é o que chamamos de “half duplex”.

Várias rotinas foram criadas, para garantir robustez ao sistema, por exemplo, caso atinja um certo tempo sem receber mensagem o próprio dispositivo reinicia para caso o problema seja de inicialização. O caso mais comum disso é quando a ESP32 Slave estava conectada a rede, mas por algum motivo a ESP32 Master perde essa conexão, sendo necessário reiniciar a rede hospedada pela Master, para que a Slave consiga se conectar novamente.

Em continuação com a comunicação sem fio, o próximo meio realizado é a comunicação por bluetooth, mais precisamente BLE (Bluetooth Low Energy). É a última versão de comunicação via bluetooth que se baseia em baixa consumo de energia.

Por conta de ser uma comunicação bem mais sofistica, não existe tanto suporte de biblioteca em comparação com a comunicação via wifi. A estrutura se baseia muito em uma estrutura de dados hierárquico, como mostra a figura 1.

**Figura 1 – Estrutura de Dados Hierárquica da Comunicação BLE**



Fonte: <https://randomnerdtutorials.com/esp32-ble-server-client/>

Como mostra a figura 1, a primeira camada é o profile, que é basicamente o que engloba um conjunto de serviços. Por sua vez, os serviços é o conjunto de características que vai ter o sistema. Por fim, as características é que pode ser interpretado como os dados que realmente vão ser enviados.

Em resumo, um serviço normalmente vai ter diversas características, no exemplo da figura 1, o serviço irã informar dois dados, um relacionado a temperatura recolhida pelo sensor, e o segundo pela humidade recolhida pelo sensor. Cada característica possui três informações importantes: "properties", "value" e "descriptor". Em resumo, "propertier" vai informar como os valores vão se interagir entre os dispositivos, algumas das propriedades possíveis são: read, write, notify, broadcast, indicate. "descriptor" é os metadados, e o "value" nada mais é que os dados que serão enviados entre os dispositivos.

Outra informação muito importante, é que cada característica e cada serviço precisa de um UUID (Universally Unique Identifier), nada mais é que um

ID para identificar o serviço e a características que pretende se conectar. Esse ID é gerado a partir do site: <https://www.uuidgenerator.net/>

Para a nossa aplicação, foi preciso criar um serviço, com duas características, a primeira para enviar os dados, e a segunda para receber os dados, dessa forma se cria uma comunicação full duplex, podendo enviar os dados de ambas direções ao mesmo tempo.

Como a utilização dessa comunicação é complexa, está sendo desenvolvida uma biblioteca auxiliar, de modo a não precisar que o programador saiba como manipular classes de função de call-back para conseguir programar essa comunicação, tornando-a algo parecido com o que se tem para comunicação via wifi.

Voltando agora para comunicação serial entre a ESP32 e a Raspberry, está sendo desenvolvida uma certa robustez para essa comunicação, de um modo bem similar com o que foi realizado para comunicação wifi.

Em paralelo, o professor Fabiano pediu para fazer alguns testes com a comunicação wifi do próprio Raspberry, para então fazer a comparação entre a comunicação ESP32 – ESP32 com Raspberry – Raspberry. O intuito é comparar o alcance máximo, visto que a antena da ESP32 é melhor, além de ser possível colocar uma antena auxiliar, assim como testar a taxa de comunicação entre elas.